Gen3 Webinar
NCIDCFS
Interoperability with Framework Services
Tuesday, June 9, 2020
2:00 PM - 3:00 PM (CST)
Interoperability with Framework Services

Chris Meyer
Jiaqi Liu
Alexander VanTol

Center for Translational Data Science,
University of Chicago
Data commons co-locate data, storage and computing infrastructure with commonly used software services, tools & apps for analyzing and sharing data to create a resource for the research community.

The Framework Services is a set of interoperable software services with public APIs that enable data commons and compute environments to receive, manage and share structured clinical data and object data in a secure and scalable way.

1. Identify Data through persistent Digital IDs that remain unchanged regardless of the physical location of your data
2. Expose data through an API
3. Expose the data model through an API
4. Interoperate with third party authN and authZ services from trusted platforms
5. Interoperate with other trusted resources with similar security and compliance.
Gen3 Implementation of Framework Services
Framework Services Architecture

Policy Engine (Arborist)

AuthN/AuthZ (Fence)

User

Metadata

New

REST

Data Indexing (Indexd)

Cloud Storage
Standards Used by NCI DCFS
Where do Standards Come From

Global Alliance
for Genomics & Health
Collaborate. Innovate. Accelerate.
GA4GH Data Repository Service (DRS)
Indexd

Gen3 data indexing service
Indexd

Gen3 data indexing

Indexing: locate data with easily used identifiers
Gen3 Implementation in Indexd
Support New DRS endpoints in Indexd

BasePath: /ga4gh/drs/v1

Schemes: HTTPS

5.1. Get info about a **DrsObject**.

GET /objects/{object_id}

5.2. Get a URL for fetching bytes.

GET /objects/{object_id}/access/{access_id}
Fetching Data using Signed URLs

- Indexd will location of file object with additional file metadata in the 
  /objects/{object_id} endpoint (open access)

- For signed URLs:
  - Users will get an OAuth2.0 access token from Fence
  - Users will auth with an OAuth2.0 access token in the header
  - Indexd will return a signed URL in
    /object/{object_id}/access/{access_id} with proper authorization
  - If user is not authorized to access data, Indexd will return access denied

Request

GET /ga4gh/drs/v1/objects/{GUID}

Authorization: Bearer <access token>
Example DRS Response for Single File Object (DRSObject)

```json
{
  access_methods: [
    {
      access_id: "gs",
      access_url: {
        url: "gs://gdc-tcga-pha00178-controlled-staging/tcga/RNA/RNA-Seq/UNC-LCCC/ILLUMI
SN749_0051_AB0168ABXX_4.tar.gz"
      },
      region: "",
      type: "gs"
    },
    {
      access_id: "s3",
      access_url: {
        url: "s3://tcga-protected-dcf-databucket-gen3/testdata"
      },
      region: "",
      type: "s3"
    }
  ],
  aliases: [],
  checksums: [
    {
      checksum: "2edd5f6db4f1deac4ef2bdf969de9f8ad",
      type: "md5"
    }
  ],
  contents: [],
  description: "",
  id: "0027045b-9ed6-45af-a68-e-f55037b5184c",
  mime_type: "application/json",
  name: null,
  size: 6703858793,
  version: "7235f205"
}
```
Example Requests: Access Endpoint (Signed URLs)

Request

1 GET /ga4gh/drs/v1/objects/{GUID}/access/{access_id}
2 Authorization: Bearer <access token>

Response Object

1 {
2     "url": "string", // SIGNED URL
3 }
Coming Soon: Bundles

- A Data Bundle is like a folder - contains a collection of data objects (can also contain other bundles)
- Support Bundles as new object type in Indexd
- Support expansion of Bundles in ContentObjects array per DRS spec

Bundle 1
  +- Object 1
  +- Object 2

Bundle 2
  +- Object 3
  +- Bundle 3
     +- Object 4
     +- Object 5
  +- Bundle 4
     +- Object 6
     +- Object 7
Future DRS usage: Interoperating with Clinical Data
GA4GH Passports & Visas
Passports & Visas

What is a Passport?

- An identity that travels with the researcher across data platforms
- A collection of visas

What is a Visa?

- An assertion signed by a visa issuer
- Designed for machine interpretation only
Behind the Curtain: JWTs

- Cryptographically signed by fence
  - Use tokens for authentication
  - Any service can verify that a token was issued by the fence instance it expects
- Contains user information
  - User tokens for authorization
- Open source libraries for working with JWTs
  - jwt.io for list of all libraries
  - We use:
    - github.com/mpdavis/python-jose
    - github.com/jpadilla/pyjwt

```json
{
  "sub": "7",
  "azp": "test-client",
  "pur": "access",
  "aud": ["openid", "user"],
  "context": {
    "user": {
      "is_admin": false,
      "name": "test",
      "projects": {
        "test": ["read", "create", "upload"]
      }
    }
  },
  "iss": "https://portal.occ-data.org/",
  "jti": "2e6ade06-5afb-4ce7-9ab5-e206225ce291",
  "exp": 1516983302,
  "iat": 1516982102
}
```
GA4GH Passport

PASSPORT

"iss": ...
... — Passport Broker signature

PASSPORT CLAIM

"ga4gh_passport_v1": [

PASSPORT VISA(s)

"iss":
"sub":
... "ga4gh_visa_v1": {

PASSPORT VISA OBJECT

"type": PASSPORT VISA TYPE
"asserted": ...
"value": ...
"source": ...
...

} — Passport Visa Issuer signature

]
When Interoperating with Visa issuers to compile information about a user’s access, Fence will be a Passport **Broker**

By interpreting and enforcing the authz information in Visas, Fence will act as a Passport **Clearinghouse**

Source of Image: [GA4GH DURI Passport overview](https://gen3.datacommons.org/duri/passport-overview)
OIDC & OAuth 2.0
What is OAuth2?

OAuth2 is a protocol allowing an application to securely access a resource on behalf of a user.
What is OpenID Connect (OIDC)?

(Identity + Authentication) + OAuth2.0 = Open ID Connect

- Authentication Layer on top of OAuth2.0
- Enables secure interoperability across systems
Overview of OAuth2 & OpenID Connect

Flow goes this way

Fence
- Authenticate user and ask to grant access to client
- Redirect back to client with code
- Create tokens and return them to the client

Client
- Send initial auth request
- Receive code
- Send token request, using the code
- Get back tokens:
  - ID (for user)
  - Access
  - Refresh
AuthN Provider

G Suite

Google (Personal)

ORCID

InCommon

elinx

Microsoft

eRA

eduGAIN

Framework Services

RAS

Fence
AuthZ Provider

Arborist

Fence

User Sync

REST

user.yaml

dbGaP Telemetry Files
Metadata API

A Framework Services API that allows clients to query and retrieve schema-less JSON blobs for GUIDs
Metadata

Current
- Indexd (persistent identifier service)
  - File name
  - File size
  - Checksum
  - URLs/locations

New
- Metadata API
  - Other arbitrary metadata

Requirements for metadata:
- Publically available data
- Available fully programmatically from a stable API
  - Not manually curated
- Schema-less
  - Cannot enforce restrictions on format
Metadata API

- API for retrieving schema-less JSON metadata blob for GUIDs

```
{
  "guid_type": "indexed_file_object",
  "dbgap": {
    "submitted_sample_id": "93227",
    "submitted_subject_id": "93227",
    "biosample_id": "SAMN08666480",
    "biosample_sample_id": "2957886",
    "sra_sample_id": "SR553389514",
    "submitted_subject_id": "93227",
    "study_subject_id": "phs001554.v1.93227",
    "dbgap_subject_id": "2474022",
    "sra_data_details": {
      "status": "public",
      ...
    },
    "study": "phs001554",
    "study_accession": "phs001554.v1.1",
    "study_accession_with_consent": "phs001554.v1.p1",
    "((/non dbgap data source))": {
      "key": "value",
    }
  },
  "{{non dbgap data source}}": {
    "key": "value",
  }
}
```
The Framework Services mean any collection of services that implements the APIs in the Framework Services API Collection.
● [github.com/uc-cdis](http://github.com/uc-cdis)

● [Gen3.org](http://Gen3.org)

● [Gen3 Community on Slack](http://Gen3 Community on Slack)

● [dcf-support@datacommons.io](http://dcf-support@datacommons.io)

● [ctds.uchicago.edu](http://ctds.uchicago.edu)
Selected Data Commons Using Gen3

NIH | NATIONAL CANCER INSTITUTE
Cancer Research Data Commons

BioData | CATALYST
Powered by Gen3

NIAID | DATA HUB PILOT

AnVIL